

LAPORAN RESMI
PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK
POLYMORPHISM



Fadilah Fahrul Hardiansyah S.ST., M. Kom

Ratri Maria Manik

3121600039

D4 TEKNIK INFORMATIKA – B

PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

TA 2022/2023

a. Class Fans

```
public class Fans{
    private String name;

    public Fans(){
        name = "noname";
    }

    public Fans(String name){
        this.name = name;
    }

    public void showName(){
        System.out.print("\n" + name + " : ");
    }

    public void watchingPerformance(){
        System.out.println(this.name + " : " + "melihat idolanya dari
youtube");
    }

    public void watchingPerformance(Musician musician){
        showName();
        System.out.print("melihat idolanya");
        musician.perform();
    }
}
```

b. Class KpopFans

```
public class KpopFans extends Fans{

    public KpopFans(){
        super();
    }

    public KpopFans(String name){
        super(name);
    }

    public void watchingPerformance(Musician musician, String expression){
        super.showName();
        System.out.print(expression + " melihat idolanya");
        musician.perform();
    }
}
```

c. Class Musician

```
public class Musician{  
    public void perform(){  
        System.out.print("Beraksi di atas panggung");  
    }  
}
```

d. Class Singer

```
public class Musician{  
    public void perform(){  
        System.out.print("Beraksi di atas panggung");  
    }  
}
```

e. Class Kpop

```
public class Kpop extends Singer{  
    public void perform(){  
        super.perform();  
        System.out.print(", dan ngedance");  
    }  
}
```

f. Class Biduan

```
public class Biduan extends Singer{  
    public void perform(){  
        super.perform();  
        System.out.print(", dengan cengkok melayu");  
    }  
}
```

g. Class Test

```
public class Test {  
    public static void main(String args[]){  
        Fans fans1 = new Fans();  
        Fans fans2 = new Fans("Mona");  
        Fans fans3 = new KpopFans("Tomi");  
        KpopFans fans4 = new KpopFans("Febi");  
  
        fans1.watchingPerformance();  
        fans2.watchingPerformance(new Musician());  
        fans2.watchingPerformance(new Singer());  
        fans3.watchingPerformance(new Biduan());  
        fans4.watchingPerformance(new Kpop(), "teriak histeris");  
    }  
}
```

h. Hasil Kompilasi

```
C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>
C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Fans.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac KpopFans.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Musician.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Singer.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Kpop.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Biduan.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>javac Test.java

C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>java Test
noname : melihat idolanya dari youtube

Mona : melihat idolanyaBeraksi di atas panggung
Mona : melihat idolanyaBeraksi di atas panggung, bernyanyi dengan merdu
Tomi : melihat idolanyaBeraksi di atas panggung, bernyanyi dengan merdu, dengan cengkok melayu
Febi : teriak histeris melihat idolanyaBeraksi di atas panggung, bernyanyi dengan merdu, dan ngedance
C:\Users\ratri\OneDrive\Documents\OOP\Praktikum 14>
```

LATIHAN

1. Tunjukkan contoh overloading dalam percobaan diatas!

a. Overloading Konstruktor

```
public class Fans{
    private String name;

    public Fans(){
        name = "noname";
    }

    public Fans(String name){
        this.name = name;
    }
}
```

```
public class KpopFans extends Fans{

    public KpopFans(){
        super();
    }

    public KpopFans(String name){
        super(name);
    }
}
```

b. Overloading method

```
public void watchingPerformance(){
    System.out.println(this.name + " : " + "melihat idolanya dari
youtube");
}

public void watchingPerformance(Musician musician){
    showName();
    System.out.print("melihat idolanya");
    musician.perform();
}
```

2. Tunjukkan contoh overriding method dan overridden method pada percobaan di atas

a. Overriding method

```
public class Singer extends Musician{
    public void perform(){
        super.perform();
        System.out.print(", bernyanyi dengan merdu");
    }
}
```

```
public class Kpop extends Singer{
    public void perform(){
        super.perform();
        System.out.print(", dan ngedance");
    }
}
```

```
public class Biduan extends Singer{
    public void perform(){
        super.perform();
        System.out.print(", dengan cengkok melayu");
    }
}
```

b. Overriden method

```
public class Musician{
    public void perform(){
        System.out.print("Beraksi di atas panggung");
    }
}
```

3. Tunjukkan contoh overloading yang terjadi dalam satu class, pada percobaan di atas

```
public void watchingPerformance(){
    System.out.println(this.name + " : " + "melihat idolanya dari
youtube");
}

public void watchingPerformance(Musician musician){
    showName();
    System.out.print("melihat idolanya");
    musician.perform();
}
```

4. Tunjukkan contoh overloading yang terjadi antara superclass dan subclass pada percobaan diatas (Fans dan KpopFans)

```
public void watchingPerformance(){
    System.out.println(this.name + " : " + "melihat idolanya dari
youtube");
}

public void watchingPerformance(Musician musician, String expression){
    super.showName();
    System.out.print(expression + " melihat idolanya");
    musician.perform();
}
```

5. Tunjukkan contoh virtual method invocation yang terjadi pada percobaan di atas

```
fans2.watchingPerformance(new Musician());
fans2.watchingPerformance(new Singer());
fans3.watchingPerformance(new Biduan());
fans4.watchingPerformance(new Kpop(), "teriak histeris");
```

6. Tunjukkan contoh polimorfism pada percobaan di atas

```
Fans fans3 = new KpopFans("Tomi");
```

7. Tunjukkan contoh inheritance pada percobaan di atas

```
public class KpopFans extends Fans
public class Singer extends Musician
public class Biduan extends Singer
public class Kpop extends Singer
```